6 7 8 1 2 2

CLAIMS

What is claimed is:

1	1.	A computer system, comprising:
2		a host bridge;
3	•	a plurality of CPUs coupled to said host bridge;
4		a system resource coupled to said host bridge; and
5		an output device coupled to said CPUs;
6		wherein said host bridge includes storage for CPU task priorities, each CPU being capable
7		of informing the host bridge of its task priority, and said host bridge uses said task
8		priorities when deciding how to allocate said system resource to said CPUs.

- 2. The computer system of claim 1 wherein said storage in said host bridge includes a table in which said host bridge stores said task priorities.
- 1 3. The computer system of claim 2 wherein said table includes an entry for each of said 2 CPUs, a task priority for a CPU being stored in the entry corresponding to that CPU.
- 1 4. The computer system of claim 1 wherein each of said CPUs transmit its task priority to said
- 2 host bridge via a cycle on a bus interconnecting said CPU and said host bridge.
- 1 5. The computer system of claim 4 wherein said cycle also includes a request by the CPU for access to said system resource.

- 1 6. The computer system of claim 4 wherein said cycle is separate from a cycle in which said
- 2 CPUs request access to said system resource.
- 1 7. The computer system of claim 1 wherein said system resource comprises memory.
- 1 8. The computer system of claim 1 wherein said system resource includes a peripheral device
- 2 coupled to said host bridge.
 - 9. The computer system of claim 1 wherein said host bridge uses said task priorities as the sole criterion for deciding how to allocate said system resource.
- 1 10. The computer system of claim 1 wherein said host bridge decides how to allocate said
- 2 system resource based on said task priorities and based on an anti-starvation algorithm.
- 1 11. The computer system of claim 1 wherein said host bridge decides how to allocate said
- 2 system resource based on said task priorities and based on a tie breaking algorithm that is used
- 3 when two or more CPUs have the highest, yet equal, task priority.
- 1 12. The computer system of claim 1 wherein said host bridge decides how to allocate said
- 2 system resource based on an algorithm that does not involve said task priorities, but uses said task
- 3 priorities to decide the resource allocation when the non task priority-based algorithm is unable to
- 4 decide between competing CPU requests for the system resource.

56854.02/1662.50400 - 13 -

- 1 13. The computer system of claim 1 wherein said host bridge decides how to allocate said
- 2 system resource based on said task priorities and based on other criteria.
- 1 14. A computer system, comprising:
- a switch; and
- a plurality of nodes coupled to said switch;
- 4 wherein said switch receives messages from said nodes, one or more of said messages
- 5 including a priority value, and said switch routes the messages based on said
- 6 priority values.
- 1 15. The computer system of claim 14 wherein said switch uses said priority values as the sole
- 2 criterion for deciding how to route said message.
- 1 16. The computer system of claim 14 wherein said switch decides how to route said messages
- 2 based on said priority values and based on an anti-starvation algorithm.
- 1 17. The computer system of claim 14 wherein said switch decides how to route said messages
- 2 based on said priority values and based on a tie breaking algorithm that is used when messages
- 3 from two or more nodes have the highest, yet equal, priority value.
- 1 18. The computer system of claim 14 wherein said switch decides how to route said messages
- 2 based on an algorithm that does not involve said priority values, but uses said priority values to

56854.02/1662.50400 - 14 -

- decide how to route said messages when the non priority value-based algorithm is unable to decide
- 4 between competing node messages.
- 1 19. The computer system of claim 14 wherein said switch decides how to route said messages
- 2 based on said priority values and based on other criteria.
- 1 20. A method of arbitrating for access to system resources, comprising:
- 2 (a) receiving a plurality of cycle requests from a plurality of CPUs;
- 3 (b) receiving task priorities associated with each of said CPUs; and
- 4 (c) granting access to a system resource based, at least in part, on said task priorities.
- 1 21. The method of claim 20 wherein (c) includes using task priorities as the sole criterion for
- 2 deciding how to grant access to a system resource.
- 1 22. The method of claim 20 wherein (c) includes using said task priorities and an anti-
- 2 starvation algorithm to grant access to the system resource.
- 1 23. The method of claim 20 wherein (c) includes granting access also based on a tie breaking
- 2 algorithm that is used when two CPUs have equal task priorities.
- 1 24. The method of claim 20 wherein (c) includes granting access based on an algorithm that
- 2 initially does not involve said task priorities, but uses said task priorities when the non task
- 3 priority-based algorithm is unable to how to grant access.

56854.02/1662 50400 - 15 -

- 1 25. The method of claim 20 wherein (c) also includes granting access based on other criteria.
- 1 26. The method of claim 20 wherein said system resource includes memory.
- 1 27. The method of claim 20 wherein said system resource includes a CPU.